

AMENDMENTS TO THE CLAIMS

Upon entry of this amendment, the following listing of claims will replace all prior versions and listings of claims in the pending application.

IN THE CLAIMS

Please amend the pending claims to add claims 40-67 as follows:

1. (original) A table lookup method, comprising:
providing to a graphical block diagram model a graphical block that defines a lookup table and having inputs for receiving input data; and
using the graphical block to update content stored in the lookup table based on received input data.
2. (original) The method of claim 1, wherein the graphical block includes block inputs, the block inputs including block input ports for connection to at least one other graphical block in a graphical block diagram model so that the input data can be received by the graphical block at the input ports from the at least one other graphical block.
3. (original) The method of claim 1, wherein the graphical block includes a block output, the block output including a block output port for connection to at least one other graphical block in a graphical block diagram model, and wherein using comprises reproducing results of the content updating at the output port.
4. (original) The method of claim 1, wherein the lookup table is used to capture the behavior of a plant, and the received input data comprises the input data and output data of the plant.
5. (original) The method of claim 4, wherein the plant input data comprises a value for at least one plant input.
6. (original) The method of claim 4, wherein the plant output data is measured data.

7. (original) The method of claim 6, wherein the plant input data and output data are received from the plant in real-time.
8. (original) The method of claim 6, wherein the plant input data and output data are read from a storage device.
9. (original) The method of claim 4, wherein the block diagram model is a model of a plant that includes a lookup table, and the plant output data is simulated data produced when the model is executed.
10. (original) The method of claim 2, further comprising:
maintaining the graphical block in a block library; and
instantiating the graphical block to create the lookup table.
11. (original) The method of claim 10, further comprising receiving parameters from a user to instantiate the graphical block.
12. (original) The method of claim 11, wherein receiving comprises providing the user with a dialog box for specifying values of the parameters of the graphical block.
13. (original) The method of claim 12, wherein the parameters comprise breakpoint data and initial table data.
14. (original) The method of claim 2, further comprising receiving block parameters from a user to initialize the block, the parameters including breakpoint data and initial table data.
15. (original) The method of claim 14, wherein the graphical block defines an adaptation process and wherein using comprises enabling the adaptation process to update the table content according to a statistical estimation algorithm.
16. (original) The method of claim 15, wherein the plant input data is usable by the graphical block to determine a location in the lookup table.
17. (original) The method of claim 16, wherein the location corresponds to a table cell.

18. (original) The method of claim 17, wherein the table content to be updated is a value associated with the table cell.
19. (original) The method of claim 15, wherein the adaptation process comprises a Recursive Sample Mean algorithm.
20. (original) The method of claim 15, wherein the adaptation process comprises a Recursive Sample Mean algorithm and uses a forgetting factor to place more weight on most recently received plant output values.
21. (original) The method of claim 17, wherein the location corresponds to a table cell and a particular operating point within the table cell, the location of the operating point within the cell being determined by interpolation of neighboring cell points.
22. (original) The method of claim 21, wherein the table content to be updated comprises values of neighboring points used in the interpolation and wherein the adaptation process weights the updated values based according to results of the interpolation.
23. (original) The method of claim 22, wherein the adaptation process comprises a Least Mean Squares algorithm.
24. (original) The method of claim 22, wherein using comprises determining an adapted operating point by interpolation using the values of the neighboring points after such points have been updated by the adaptation process.
25. (original) The method of claim 1, wherein using comprises executing the graphical block diagram model for simulation purposes.
26. (original) A method of claim 1, wherein using comprises:
initializing the lookup table with initial table data and breakpoint data;
using the graphical block diagram of the graphical block diagram model as a specification for interpretation by automatic code generation software that generates code to

perform computations equivalent to the computations performed by the graphical block diagram model; and

executing the generated code in a controller of a real-time control application.

27. (original) The method of claim 1, wherein the lookup table is included in a simulation model that simulates an embedded control system.

28. (original) The method of claim 1, wherein breakpoint data specified by a user partitions the table into cells and the content that is updated is a cell value, further comprising:

adjusting the spacing of the breakpoint data to control the number of cells in the table.

29. (original) The method of claim 2, wherein the block inputs further include an adaptation control signal to enable or disable the lookup table content updating.

30. (original) The method of claim 2, wherein the block inputs are configured to connect to a locking mechanism that restricts the content updating to a particular cell in the lookup table.

31. (original) The method of claim 3, wherein the block output further includes a copy of the lookup table content at all table locations after the updating is completed.

32. (original) The method of claim 3, wherein the block output includes a table index number corresponding to the location of the updated content.

33. (original) A computer program product residing on a computer-readable medium that provides a table lookup, the computer program product comprising instructions causing a computer to:

provide to a graphical block diagram model a graphical block that defines a lookup table and having inputs for receiving input data; and

use the graphical block to update content stored in the lookup table based on received input data.

34. (original) A method, comprising:

operating a lookup table in a static lookup mode in which the table receives one or more input values, determines at least one previously stored value corresponding to the input values, and produces output data using the at least one previously stored value; and

operating the lookup table in a dynamic adaptation mode in which the table receives the at least one input value and at least one additional value, and in which the table uses the at least one additional value to modify previously stored at least one value corresponding to the at least one input value.

35. (original) The method of claim 34, wherein, in the dynamic adaptation mode, the table produces output data using the modified previously stored at least one value.

36. (original) The method of claim 34, wherein the lookup table is defined by a block that is part of a block diagram model.

37. (original) The method of claim 34, wherein the lookup table is operated in the dynamic adaptation mode in an interpreted block diagram environment, and further comprising, after operating the lookup table in the dynamic adaptation mode, generating compiled code using the modified previously stored at least one value, and using the compiled code to operate the lookup table in the static lookup mode.

38. (original) The method of claim 34, wherein the at least one additional value is received from a sensor, and further comprising, during the dynamic adaptation mode, determining whether the sensor has failed and switching to the static mode if the sensor has failed.

39. (original) The method for claim 34, comprising:

operating the lookup table in the static mode;

during operation in the static mode, switching to dynamic mode to update the previously stored at least one value; and

switching back to static mode after the previously stored at least one value has been updated.

40. (new) In a device, a system for providing a lookup table, the system comprising:
a graphical block associated with a graphical block diagram model, the graphical block defining a lookup table and having inputs for receiving input data; and
wherein the graphical block includes an updating mechanism to update content stored in the lookup table of the graphical block based on received input data.
41. (new) The system of claim 40, wherein the graphical block includes block inputs, the block inputs including block input ports for connection to at least one other graphical block in a graphical block diagram model so that the input data can be received by the graphical block at the input ports from the at least one other graphical block.
42. (new) The system of claim 40, wherein the graphical block includes a block output, the block output including a block output port for connection to at least one other graphical block in a graphical block diagram model, and wherein the graphical block is used to reproduce results of the content updating at the output port.
43. (new) The system of claim 40, wherein the lookup table is used to capture the behavior of a plant, and the received input data comprises the input data and output data of the plant.
44. (new) The system of claim 43, wherein the plant output data comprises measured data.
45. (new) The system of claim 44, wherein the plant input data and output data are received from the plant in real-time.
46. (new) The system of claim 44, wherein the plant input data and output data are read from a storage device.
47. (new) The system of claim 44, wherein the block diagram model is a model of a plant that includes a lookup table, and the plant output data is simulated data produced when the model is executed.
48. (new) The system of claim 41, wherein the graphical block is maintained in a block library, and the graphical block is instantiated to create the lookup table.

49. (new) The system of claim 41, wherein block parameters are received from a user to initialize the block, the parameters including breakpoint data and initial table data.
50. (new) The system of claim 49, further comprising a dialog box for specifying values of the parameters of the graphical block.
51. (new) The system of claim 48, wherein the graphical block defines an adaptation process and is used to enable the adaptation process to update the table content according to a statistical estimation algorithm.
52. (new) The system of claim 51, wherein the plant input data is used by the graphical block to determine a location in the lookup table.
53. (new) The system of claim 52, wherein the location corresponds to a table cell, and the table content is updated is a value associated with the table cell.
54. (new) The system of claim 51, wherein the adaptation process comprises a Recursive Sample Mean algorithm and uses a forgetting factor to place more weight on most recently received plant output values.
55. (new) The system of claim 54, wherein the location corresponds to a table cell and a particular operating point within the table cell, the location of the operating point within the cell being determined by interpolation of neighboring cell points.
56. (new) The system of claim 55, wherein the table content to be updated comprises values of neighboring points used in the interpolation and wherein the adaptation process weights the updated values based according to results of the interpolation.
57. (new) The system of claim 56, wherein the adaptation process comprises a Least Mean Squares algorithm.
58. (new) The system of claim 56, wherein the graphical block determines an adapted operating point by interpolation using the values of the neighboring points after such points have been updated by the adaptation process.

59. (new) The system of claim 40, wherein the graphical block diagram model is executed for simulation purposes.

60. (new) A system of claim 40,
wherein the system further comprises automatic code generation software and a controller of real-time controller application;
wherein the lookup table is initialized with initial table data and breakpoint data;
wherein the graphical block diagram of the graphical block diagram model is used as a specification for interpretation by the automatic code generation software that generates code to perform computations equivalent to the computations performed by the graphical block diagram model; and
wherein the generated code is executed in the controller of the real-time control application.

61. (new) The system of claim 40, wherein the lookup table is included in a simulation model that simulates an embedded control system.

62. (new) The system of claim 40,
wherein breakpoint data specified by a user partitions the table into cells and the content that is updated is a cell value; and
wherein the spacing of the breakpoint data is adjusted to control the number of cells in the table.

63. (new) The system of claim 41, wherein the graphical block inputs further include an adaptation control signal to enable or disable the lookup table content updating.

64. (new) The system of claim 41, wherein the graphical block inputs are configured to connect to a locking mechanism that restricts the content updating to a particular cell in the lookup table.

65. (new) The system of claim 42, wherein the graphical block output further includes a copy of the lookup table content at all table locations after the updating is completed.

66. (new) The system of claim 43, wherein the graphical block output includes a table index number corresponding to the location of the updated content.

67. (new) The system of claim 51, wherein the adaptation process comprises a Recursive Sample Mean algorithm.